Experiment 7:

1. Check mvn install and set path 🡪 in evirement 🡪 in Jenkins-tools
2. Make folder in desktop 🡪go cmd 🡪past following code

mvn archetype:generate -DgroupId=com.example -DartifactId=JavProject -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

1. Make folder name as chromedriver 🡪 exact this file https://chromedriver.storage.googleapis.com/114.0.5735.90/chromedriver\_win32.zip
2. Go to vs code 🡪 open folder that created at step2 🡪 in pom.xml and below dependency past following code:
3. <dependency>

<groupId>org.seleniumhq.selenium</groupId>

<artifactId>selenium-java</artifactId>

<version>3.141.59</version>

</dependency>

1. Install extension – **Extension Pack for Java** , **mvn-package-auto** 🡪In src make new file name as : SeleniumExample.java

package com.example;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class SeleniumExample{

public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver","C:\\Users\\USER\\Downloads\\chromedriver\_win32"); //step 3 folder path copy and paste here

WebDriver driver = new ChromeDriver();

driver.get("https://www.example.com");

driver.quit();

}

1. Go to step2 folder – javaproject - cmd
2. mvn clean
3. mvn test
4. go to github 🡪 make new repository 🡪 click on add – upload 🡪 step 2 folder >javproject> select all file 🡪 target and upload in github 🡪 copy code
5. go to Jenkins 🡪 manage jenkins 🡪 plugins install
6. selenium
7. selenium builder
8. chromedriver

check in available plugins 🡪 tesrNG

1. in Jenkins 🡪 new items – name -mavenproject

general 🡪source code manage - git -url paste 🡪branch to build - \*/main 🡪 goal – test 🡪save

1. build project

experiment: 06

1. tomcat 3.9.5 and second link of zip file 🡪 set user name , password and port
2. open Jenkins 🡪 java -jar Jenkins.war httpPort=8080 and tomcat 🡪startup
3. go to file explorer 🡪 in c drive 🡪 program file 🡪 apache Software Foundation 🡪 Tomcat 9.0 🡪 conf 🡪 tomcat-user.xml open 🡪 change in roles="manager-gui, manager-script"
4. go to Jenkins 🡪 manage Jenkins 🡪 plugins – available -war- deploy container
5. go to Jenkins 🡪 manage Jenkins 🡪 security 🡪 credential 🡪 global 🡪 tomcat -user, passaword ,name
6. in dashboard of Jenkins 🡪 new item 🡪 name , maven project
7. general 🡪 git project and sorcecode – git add <https://github.com/ashokitschool/JAVA-MAVEN-WEB-APP>
8. build triger 🡪 build when snapshot , poll scm \* \* \* \* \* , ingnor postcom
9. goal 🡪 install
10. build env. 🡪 delete env.
11. add post build action 🡪 deploy war/er cont. 🡪 tomcat9.x 🡪 credentioal \*\*/\*.war, sample name ,add cridentioal
12. apply and save and build
13. file manager 🡪 tomcat🡪webapps 🡪show --sample.war

experiment 1 and 2

1. open new folder and open git bash
2. git –version
3. git config --global user.name “”
4. git config --global user.email “”
5. git config --global user.password “”
6. git config –list
7. git init
8. ls -a
9. git clone any repository (https://github.com/Aashlesha07/newgit.git)
10. git status
11. git add . or git add –a
12. git commit -m “add”
13. git branch hello
14. git checkout hello
15. touch index.html
16. git add .
17. git checkout master
18. git status
19. git branch
20. git remote add origin any (https://github.com/Aashlesha07/newgit.git)
21. git branch -M main
22. git branch
23. git pust -u origin main
24. git status
25. git log
26. git log --oneline

Experiment 8

